

PROVISIONAL APPLICATION FOR PATENT

under

37 CFR §1.53(c)

09512225-032301

TITLE: COMPUTATION OF INTRINSIC PERCEPTUAL
SALIENCY IN VISUAL ENVIRONMENTS, AND
APPLICATIONS

APPLICANT: CHRISTOF KOCH AND LAURENT ITTI

CERTIFICATE OF MAILING BY EXPRESS MAIL

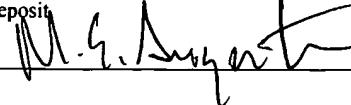
Express Mail Label No. EL688320670US

I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

July 23, 2001

Date of Deposit

Signature


Mike Augustine

Typed or Printed Name of Person Signing Certificate

**COMPUTATION OF INTRINSIC PERCEPTUAL SALIENCY IN VISUAL
ENVIRONMENTS, AND APPLICATIONS**

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority from provisional application No. 60/274,674 filed March 8, 2001, and 60/288,724 filed May 4, 2001.

06618-790001-CIT-3411

STATEMENT AS TO FEDERALLY-SPONSORED RESEARCH

[0002] This work was supported by the Engineering Research Centers Program of the National Science Foundation under Award Number EEC-9402726 and by the Office of Naval Research under Award Number N00014-95-1-0600. The US Government may have certain rights in this invention.

BACKGROUND

[0003] Different systems for analysis of vision components are known. Many of these systems, broadly categorized under machine vision, attempt to use the machine, usually a digital computer running dedicated software, to actually identify parts of the image.

[0004] However, vision algorithms frequently fail when confronted with real life images. These real life images may be of extremely high resolution, e.g., on the order of 6000 by 4000 pixels, and may be very cluttered with information that might not necessarily be relevant to the visual task at hand. For instance, many images may have partially occluding objects such as foliage, vehicles, people and so on.

[0005] It is believed that biological vision systems use a different approach. The mammalian visual system is believed to use a computational strategy of identifying interesting parts of the image without extensively analyzing the content of the image. The entire image may be analyzed in parallel for simple features. Portions of the image are then selected, based either on their behavioral relevance or based on local image cues. The local image cues may include brightness, motion, and/or color and others. The mammalian brain evolved in this manner to handle the enormous amount of information that is received from a scene. This information has been estimated as being on the order of up to 10^8 bits per second along the optic nerve, the axonal fibers that constitute the output of the retina. This may exceed what the brain is capable of fully processing and assimilating into its conscious experience.

[0006] Because of this processing strategy, only a small fraction of the information that is actually registered by the human visual system actually influences behavior. Different studies have demonstrated this in different ways. In some studies ("change blindness") (Rensink, R.A., O'Regan, J.K., and Clark, J.J. ``To see or not to see: The need for attention to perceive changes in scenes," Psychological Sci. 8:368-373, 1997) significant image changes are not actually perceived under natural viewing conditions. However, once the attention of the person is directed to these changes, they can be easily perceived. This implies that even though a part of an image might be registered by the brain, the conscious mind might not be visually aware of that part or any other in the image.

[0007] Those parts of an image which elicit a strong, rapid and automatic response from viewers, independent of the task they are trying to solve, can be referred to as being "visually salient". Two examples of such salient locations are a green object among red ones, or a vertical line among horizontal ones. The mind can direct its attention to other parts of the image, although that may require voluntary effort.

SUMMARY

[0008] The present invention describes a computer-based implementation that allows automatic detection of salient parts of image information. This may use a model which is based on the way the primate's visual system is believed to process the retinal image stream.

[0009] The application discloses the basic model, and applications of the model to various practical uses. One such use includes detection of the effectiveness of an image or temporal sequence of images in displaying their content, e.g., in advertising context. Some specific model attributes are also disclosed. A first model attribute describes higher order statistical analysis of image information to compute saliency. Another model attribute discloses detection of extended but interrupted contours within the image information that can contribute to image saliency. In another model attribute, the computation of saliency specific to moving objects in a video sequence or constantly changing image sequences is described. Another aspect relates to the improvement of computing saliency for video sequence detection, by detecting portions of the video sequence which flicker. Another relates to the usage of multiple spectral images acquired of the same scene. Another relates to the ability of the model to provide specific feedback on how to

improve the saliency of specific objects or locations in the scene.

[0010] At the basis of the invention was the original concept of a "saliency map" proposed by Koch and Ullman (Koch, C. and Ullman, S. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, **4**:219-227, 1985), and two detailed computer implementations: Itti, L., Koch, C. and Niebur, E. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. Pattern Analysis & Machine Intell. (PAMI)* **20**:1254-1259, 1998 and Itti, L. and Koch, C. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research* **40**:1489-1506, 2000.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] These and other aspects of the invention will be described in detail with reference to the accompanying drawings, wherein:

[0012] Figure 1 shows a flow diagram of a model of saliency-based attention;

[0013] Figure 2 shows a block diagram of the nonlinear filtering using an iterated difference of Gaussian filter;

[0014] Figure 3 shows a diagram of waveforms obtained at different spatial resolutions or scales;

[0015] Figure 4A - 4H shows results of different numbers of iterations of the iterative equation to converge to salient elements;

[0016] Figure 5 shows an exemplary field with a background and an internal elliptical area;

[0017] Figure 6 shows a block diagram of a statistical measure of pixel distribution using higher order statistics;

[0018] Figure 7 shows a flowchart of operation of obtaining the different image pyramids;

[0019] Figure 8 shows a diagram of the different pyramids obtained;

[0020] Figure 9 shows a flowchart of finding extended image contours;

[0021] Figures 10A-10C show additional information in finding the extended contours;

[0022] Figure 11 shows some notion of the different image contour operations;

[0023] Figure 12 shows a flowchart of motion in an extended image sequence.

[0024] Figure 13 shows a flowchart of thresholding.

DETAILED DESCRIPTION

[0025] Figure 1 shows a system for determining a saliency map, which may be a two-dimensional map that encodes salient objects in a visual environment. The map of the scene expresses the saliency of all locations in this image. This map is the result of competitive interactions among feature maps for image features including color, orientation, texture, motion, depth and so on, that interact within and across each map. At any time, the currently strongest location in the saliency map corresponds to the most salient object. The value in the map represents the local saliency of any one location with respect to its neighborhood. By default, the system directs attention towards the most salient location.

[0026] A second most salient location may be found by inhibiting the most salient location, causing the system to automatically shift to the next most salient location.

[0027] The techniques described herein are based on the bottom-up control of attention, i.e., control that is based on the properties of the visual stimulus. This compares with a top-down component, which may be based not only on the content of the image but also on additional high-level features that may depend on a specific visual task at hand. An example of a top-down

component would include, for example, storing an image of a face of a person one is searching for, followed by correlating that image across the entire scene.

[0028] A task of the saliency map is to compute a scalar quantity representing the salience at every location in the visual field, and to guide the subsequent selection of attended locations. The "feature maps" provide the input to the saliency map, which is modeled as a neural network receiving its input at a particular spatial scale (here scale 4).

[0029] The input image 100 may be a digitized image from a variety of sources. In one embodiment, the digitized image may be from an NTSC video camera.

[0030] At 105, linear filtering is carried out at different spatial scales, here nine spatial scales. The spatial scales may be created using Gaussian pyramid filters of the Burt and Adelson type. These pyramid filters may include progressively low pass filtering and sub-sampling of the input image. The spatial processing pyramids can have an arbitrary number of spatial scales. In the example provided, nine spatial scales provide horizontal and vertical image reduction factors ranging from 1:1 (level 0, representing the original input image) to 1:256 (level 8) in powers of 2. This may be used to detect differences in the image between fine and coarse scales.

TOP SECRET//NOFORN

[0031] Each portion of the image is analyzed by comparing the "center" portion of the image with the surround part of the image. Each comparison, called "center-surround" difference, may be carried out at multiple spatial scales indexed by the scale of the center, c , where, for example, $c = 2, 3$ or 4 in the pyramid schemes. Each one of those is compared to the scale of the surround $s = c + d$, where, for example, d is 3 or 4 . This example would yield 6 feature maps for each feature at the scales $2-5, 2-6, 3-6, 3-7, 4-7$ and $4-8$ (for instance, in the last case, the image at spatial scale 8 is subtracted, after suitable normalization, from the image at spatial scale 4). One feature type encodes for intensity contrast, e.g., "on" and "off" intensity contrast shown as 115 . This may encode for the modulus of image luminance contrast, which shows the absolute value of the difference between center intensity and surround intensity. The differences between two images at different scales may be obtained by oversampling the image at the coarser scale to the resolution of the image at the finer scale. In principle, any number of scales in the pyramids, of center scales, and of surround scales, may be used.

[0032] Another feature 110 encodes for colors. With r, g and b respectively representing the red, green and blue channels of the input image, an intensity image I is obtained as $I =$

$(r+g+b)/3$. A Gaussian pyramid $I(s)$ is created from I , where s is the scale. The r , g and b channels are normalized by I at 131, at the locations where the intensity is at least 10% of its maximum, in order to decorrelate hue from intensity.

[0033] Four broadly tuned color channels may be created, for example as: $R=r-(g+b)/2$ for red, $G=g-(r+b)/2$ for green, $B=b-(r+g)/2$ for blue, and $Y=(r+g)/2-|r-g|/2-b$ for yellow, where negative values are set to zero). 130 computes center-surround differences across scales. Two different feature maps may be used for color, a first encoding red-green feature maps, and a second encoding blue-yellow feature maps. Four Gaussian pyramids $R(s)$, $G(s)$, $B(s)$ and $Y(s)$ are created from these color channels. Depending on the input image, many more color channels could be evaluated in this manner.

[0034] In one embodiment, the image sensor 99 that obtains the image of a particular scene is a multi-spectral image sensor. This image sensor may obtain different spectra of the same scene. For example, the image sensor may sample a scene in the infra-red as well as in the visible part of the spectrum. These two images may then be evaluated in a similar manner to that described above.

[0035] Another feature type may encode for local orientation contrast 120. This may use the creation of oriented Gabor

pyramids as known in the art. Four orientation-selective pyramids may thus be created from I using Gabor filtering at 0, 45, 90 and 135 degrees, operating as the four features. The maps encode, as a group, the difference between the average local orientation and the center and surround scales. In a more general implementation, many more than four orientation channels could be used.

[0036] In summary, differences between a "center" fine scale c and a "surround" coarser scales yield six feature maps for each of intensity contrast, red-green double opponency, blue-yellow double opponency, and the four orientations. A total of 42 feature maps is thus created, using six pairs of center-surround scales in seven types of features, following the example above. A different number of feature maps may be obtained using a different number of pyramid scales, center scales, surround scales, or features.

[0037] 130 shows normalizing the features to extract salient image location from the raw center-surround maps, and to discard inconspicuous locations. This process may be critical to the operation of the system. This operation follows the flowchart of Figure 2. At 200, each feature map is first normalized to a fixed dynamic range such as between 0 and 1. This may eliminate

feature-dependent amplitude differences that may be due to different feature extraction mechanisms.

[0038] At each step of the iteration, the map is convolved by a large difference-of-Gaussians kernel at 215 and the results are added to the center contents of the map at 210. The additional input implements the short-range excitation processes and the long-range inhibitory processes between the neighboring visual locations. The map is then half-wave rectified at 220, which may remove negative results. This makes the iterative process nonlinear, which may improve the results.

[0039] Specifically, the filter carries out

$$DOG(x, y) = \frac{c_{ex}^2}{2\pi\sigma_{ex}^2} e^{-(x^2+y^2)/(2\delta_{ex}^2)} - \frac{c_{inh}^2}{2\pi\sigma_{inh}^2} e^{-(x^2+y^2)/(2\delta_{inh}^2)} \dots (1)$$

where c_{ex} and c_{inh} are positive numbers that denote the strength of the excitatory center response and the strength of the inhibitory surround response, respectively. δ_{ex} and δ_{inh} denote the width, spatial extent or size of the associated excitatory central Gaussian or the inhibitory surround Gaussian. In eq. 1, the central Gaussian is subtracted from the surround Gaussian to obtain a so-called "Mexican-Hat" operator or "Difference-of-

Gaussian", hence leading to the name 'DoG'. This can also be seen in the central box '215' of Fig. 2.

[0040] At each iteration, the feature map M goes through the following transformation:

$$M \leftarrow |M + M * DOG - C_{inh}|_{\geq 0} \dots \dots (2)$$

Eq. 2 shows getting the new value of the image 'M' by taking the current input image in map 'M', filtering it through this 'DOG' filter, adding it to the existing Map 'M', and subtracting an inhibitory constant C_{inh} . Positive results are kept; negative results are set to zero.

[0041] Each feature map is iterated 10 times using this equation. Different numbers of iterations may be carried out, based on experience and the application domain. The local excitation is counteracted by broad inhibition from neighboring locations. This spatial interaction across the entire map may be crucial for resolving competition among salient items.

[0042] Figure 3 shows two different examples of the six center-surround receptive field types. The left part of the figure shows Gaussian pixel widths, numbered 0-8, for the 9 spatial scales used in the model example of Figure 1. Scale 0 corresponds to the original image, and each subsequent scale is coarser by a factor 2. At the coarsest scale, $\sigma=8$, almost

the entire image is blurred and only very coarse objects are visible as blobs. 300 and 302 show two examples of the six center-surround receptive field types. 300 shows the scale pair 2-5 representing the image filtered with the filter with sigma=2 being subtracted from the image filtered sigma=5. 302 shows the scale pair 4-8. The spatial competition for salience may be implemented within each of the feature maps. Each map receives input from the filtering and center surround stages.

[0043] An example of results is shown in Figures 4A-4H. Figure 4A shows the actual image, with iteration 0 (Figure 4B) showing the items that are present in Figure 4A. Figure 4C shows two iterations of the type illustrated in Fig. 2 to show that the salient features begin to emerge. This is shown in further detail in Figure 4D (iteration 4), 4E (iterations 6), 4F (iteration 8), 4G (iteration 10) and 4H (iteration 12). Figure 4G representing iteration 10 clearly shows which features are most salient, and this only becomes more evident in Figure 4H showing the result of iteration 12. Since there is not that much difference between iterations 10 and 12, in this situation it is evident that the iteration can be stopped at 10. The net effect of the iterative process in this example was to reinforce the brightest object while suppressing the darker objects, which may

embody the fact that the brightest object may be perceived as visually salient by human observers.

[0044] After normalization at 130, the feature maps for intensity, color, and orientation are summed across scales into three separate "conspicuity maps," 133 for intensity, 134 for color and 136 for orientation. Conspicuity maps for other features, such as motion or flicker, can easily be added here.

[0045] Each conspicuity map is then subjected to another 10 iterations of the iterative normalization process shown in Figure 2. The motivation for the creation of three separate channels and their individual normalization is the hypothesis that similar features compete strongly for salience, while different modalities contribute independently to the saliency map. The individual normalization may provide additional information since similar features may compete strongly for salience, while different modalities may contribute independently to the saliency maps.

[0046] This "within-feature competitive" globally promotes the most salient portions of the feature map, both within a feature, and over the whole map.

[0047] After this, at 150, linear combinations of these maps are taken to form the unique saliency map shown as 155. At any given time, the maximum of the saliency map may correspond to the

most salient stimulus, and represents the item to which the focus of attention should next be directed. Hence, at any given time, the most salient location may be determined from the maximum of the saliency map. This may be effected at 160 using a "winner take all" technique.

[0048] Different "winner take all" techniques are known. However, this system may use a two-dimensional layer of integrate and fire neurons with strong global inhibition.

[0049] The system as described might direct its focus of attention constantly to one location since the same winner would always be selected. Accordingly, the feedback shown as 165 is indicated to provide feedback from the "winner take all" array 160 to the saliency map 155. That is, after some period of variable delay, the saliency of the winning location may be transiently inhibited. This assures that the "winner take all" circuit automatically selects the next most salient location. As a consequence, attention then switches to the next most conspicuous location. This inhibition prevents a previously attended location from being attended to again within a short interval and endows the entire algorithm with a dynamic element.

[0050] The above embodiment has described comparisons between different image parts at 130 which rely on simple center-surround mechanisms. These are implemented as differences between mean

pixel values, across the different spatial scales, as described. However, in some situations this might not correctly detect any dissimilarity between the center and surround regions that may be present in the higher-order spatial statistics of the input.

[0051] Another embodiment determines higher order, e.g., second order, statistics in the image. This may be done for any of previously described purposes. For example, consider the case as shown in Figure 6, where the center and surround are two different textures with similar means but different higher-order statistics (for instance, different variances). A simple comparison of the mean pixel values between the center and surround regions would show a low saliency, while both textures may appear quite dissimilar to human observers.

[0052] An alternative embodiment described herein takes into account not only mean value differences between center and surround, but also the statistical distribution of the information.

[0053] An embodiment describes the use of second-order statistics, here the variance of pixel distribution. This technique may be used when a simple comparison of mean pixel values between center and surround regions shows a low saliency. Alternatively, this may be used for all applications of the invention.

[0054] This system may provide a statistical measure of a difference of distributions of pixel values between the center and surrounding regions.

[0055] This embodiment may assume that the pixels should be distributed in a Gaussian format. While this assumption holds for only certain kinds of images, it may still represent a better approximation than the first embodiment. However, more general statistical assumptions could also be used.

[0056] An example is shown in Figure 5. An image is shown having a background area with a texture, and an elliptical area within the other background area. An observer can easily see the elliptical area within the background in Figure 5, but the average values are more or less the same.

[0057] Figure 6 shows a block diagram of a center-surround neuronal "unit" of this embodiment. This unit is comparing two different parts 600, 605 with different textures. The unit compares the distribution of pixel values between the center 605 and surround regions 600. In the example shown, the mean pixel values are substantially identical over the center and concentric surround regions. Therefore, an operator that only considered the mean intensity in the center and subtracts that from the average intensity in the surround would obtain a value close to zero and would not find the center portion to be salient. Note that the

mean of the two Gaussian distributions in the middle plot are identical.

[0058] This embodiment takes the variance as shown. The variance of the center region 610 is higher than the variance 615 of the surround. The distributions of pixel values in center and surround are approximated two Gaussian functions. A statistical measure of similarity between those distributions (such as the Kullback divergence) may then be used to compute the response of the neuron at 620, such that identical distributions yield no neuronal response while very different distributions yield a strong response.

[0059] The mean and standard deviation may be calculated as follows, and as shown in the flowchart of Figure 7. The pixel distribution is taken in a region represented by a pixel at a given level in a multiscale image pyramid.

[0060] At 700, two different image "pyramids", are created, that is two sets of images filtered at different resolutions, of sigmas in Figure 3. Each pyramid accumulates the sum and the sum of the squares of all the pixels up to the chosen level of the pyramids. That is, at a given level n in the sum pyramid, each pixel is the sum of the pixel values x_i of the $(d_n)^2$ corresponding pixels at the base level of the pyramid, where d

is the scaling between levels in the pyramid. In the specific implementation, $d=2$.

[0061] The sum-of-squares pyramid is similar except that an image of the sum of the squares of the pixel values in the original image is used as the base of the pyramid.

[0062] This data is already calculated and stored in two pyramids. Therefore the mean and standard deviation for any pixel at level n in the pyramid can be easily calculated as

$$\mu = \frac{1}{n} \sum_i x_i$$
$$\sigma^2 = \left(\frac{1}{n-1} \right) \left[\sum_i x_i^2 + \left(\frac{1}{n^2} - \frac{2}{n} \right) \left(\sum_j x_j \right)^2 \right]$$
$$n = (d^n)^2$$

at 705.

[0063] At 710, saliency is then derived from a comparison between this mean and standard deviation for the center and surrounding regions. The saliency may use other similar measures including Euclidean distance between the mean-standard deviation pair, ideal-observer discrimination, and the Kullback J-divergence.

[0064] This higher order comparison may not only be applied to the intensity channel, but also to color opponencies and orientation-selective channels or to any other channel.

[0065] Figure 8 graphically illustrates the computation of mean and variance of pixel distribution within increasingly larger square regions, using an image pyramid architecture. From the original input image 800, two dyadic image pyramids are created. In the sum pyramid on the left, each pixel at a given level "n" contains the sum of all corresponding pixels at level 0 (the original image). In the second one (right), each pixel at level "n" contains the sum of squares of all corresponding pixels at level 0.

[0066] Another improvement may include improved detection of center-surround differences by contour identification. Detection of salient contours involves the elaboration of a subset of elongated contours in the image. Image contour detection can be done with standard image processing techniques, such as by using Canny edge-detection filtering. Several techniques have been proposed for the detection of salient contours. The present technique uses a multiscale approach which is flowcharted in Figure 9, and shown graphically in Figure 10.

[0067] At 900, contours and edges of the image are detected at multiple spatial scales using oriented Gabor filters which may be

set to take account of contours in both local neighborhoods as well as contours across the entire image. This takes note that a longer contour or edge, even if interrupted, may represent a more salient image feature than shorter image segments, even if they are continuous and non-interrupted.

[0068] In this embodiment, at any given spatial scale, neighboring locations interact such that edge elements at a given orientation which appear to form a contour reinforce each other. This provides the raw map "M" containing Gabor edge detection results at a given spatial scale, with values scaled between 0 and 1. These values are iterated as follows. At 910, the image is convolved with an excitatory filter mask, yielding a new map "E".

[0069] At 915, the value 1 is added to E.

[0070] 920 saturates values > 1.25 to avoid explosion.

[0071] The raw map M is multiplied by E at 925.

[0072] At 930, M is convolved by a difference of Gaussian filter, yielding I

[0073] At 940, a small constant k is added to I which implements a global inhibitory bias,

[0074] AT 945, I is added to M,

[0075] At 950, negative values in M are eliminated by setting them to zero.

[0076] Note that this is a non-linear process, since saturation is applied at one end, and negative values are eliminated at the other end. At 955, this non-linear process 910-950 is iterated a few times (on the order of 10 iterations), hence implementing a recurrent non-linear scheme with early termination.

[0077] Figure 10A shows parameters defining the field of influence between two nearby visual neurons, as found in typical single-spatial scale models of contour integration. The actual image 1000 is filtered by banks of orientation-selective filters 1010. These filters may approximate neuronal responses for several orientations and at several spatial scales 1020, not taking into account any interaction. Figure 10B shows characterizing interactions between distant filters according to separating distance and angles. In typical models, this may yield a "field of influence" which defines the location, preferred orientation and connection strength between a central neuron of interest and its neighbors. Figure 10C shows this field of influence. Results obtained with this technique for each map M are then combined at 960, first across spatial scales for one orientation, and then across orientations as shown in Figure 11.

[0078] Local oriented features are first extracted at multiple spatial scales and for multiple orientations (here four orientations at 0, 45, 90 and 135 degrees). The iterative competition for salience and contour integration process described in the previous figure is then applied to each resulting feature map (here represented only for one map, at the right).

[0079] The result is a single saliency map which contains not only small, localized salient objects as detected with the basic technique described with reference to Figure 1, but also extended contours if those are salient.

[0080] The above has described operation with static images. An important extension of the above operates to provide a new feature channel for detection of salience in moving images in a video sequence. The operation may use the flowchart shown in Figure 12.

[0081] At 1200, visual motion from a sequence of images (acquired, for instance, via a video camera) is extracted. This may use the Spatio Temporal Energy model that has previously been described by Adelson and Bergen. However, many other motion algorithms could also be used here. Briefly, this may apply three-dimensional (x,y,t) band-pass filters to the sequence of frames. Each filter detects motion in a given direction and at a

given speed. Note that this filter is a type of orientation filter but in space-time instead of in two-dimensional spatial plane. A bank of such filters are provided and are tuned to motion in different directions such as up, down and left and right, and at different velocities, i.e., x pixels per frame. If we assume 4 directions, and three speeds, then 12 filters per image location are required.

[0082] At 1205, this motion extraction module is applied to the luminance (Y) and chrominance (C) channels of the image at several spatial scales, yielding one "motion map" for each orientation, velocity and scale.

[0083] 1210 carries out non-linear spatial competition for salience, as described previously, with respect to each resulting motion map. That is, the motion saliency of multiple objects, moving roughly in the same direction and speed, is evaluated by the competitive and iterative process described above. Again, this step is crucial for evaluating the saliency of more than one object that moves in a similar direction and speed.

[0084] At 1215, all the maps for a given orientation and velocity (and several spatial scales) are summed into one summary map for that orientation and velocity.

[0085] At 1220, all those maps are then summed using the non-linear spatial competition process, and then at 1225, all the

summary maps are summed. The final result of the non-linear spatial competition process is obtained at 1230.

[0086] This system is used for detecting saliency in the motion channel. A nonlinear within-feature competition scheme is used to detect motion in luminance and also in chrominance in a multiscale manner. This provides one motion map for each of orientation, velocity and scale for each of luminance and chrominance.

[0087] The nonlinear spatial competition system then is used for each resulting motion map.

[0088] Another embodiment recognizes that the Adelson-Bergen or spatio-temporal image filters are specialized to pick up motion. Classic motion detectors do not respond to flicker in the image since nothing is moving in any direction. Hence, an additional filter may be added which provides a temporal derivative channel to pick up the flicker. Hence, this embodiment looks at flicker in animated sequences. This may be of particular relevance for evaluating the saliency of web-pages or marquette advertising or electronic displays with flashing LEDs.

[0089] Take an example of a light turning on and off, without moving, just flashing. This most certainly attracts attention. Yet Adelson-Bergen motion detectors do not respond to flicker,

since nothing is moving in any one particular direction. A temporal derivative channel may be used to pick up flicker and integrate the derivative into saliency. An embodiment is shown in the flowchart of Figure 13.

[0090] At 1300, the absolute value of the temporal derivative of the image intensity is computed. Since an increase in light should be as salient as a decrease in light intensity, any changes in this value, whether positive or negative, are relevant

[0091] At 1305, this absolute difference value is thresholded, and compared against the threshold. That is, if the change in image intensity is too small, it is not considered, since it might be produced by noise. Other temporal information may be calculated at 1310, such as taking the derivative of colors, e.g. the red-green or blue-yellow color channels, with respect to time. Again, the absolute value of the temporal derivative in the red-green and in the blue-yellow color channels can be considered. At 1315, a test is made to determine if the change is over the whole image. If so, then the process stops.

This is based on the recognition that flickering of the entire image may not be very salient. For example, simply turning room lights quickly on and off might not be very salient. This can be carried out using spatial competitive interactions as in the other channels. At 1320, the image portion that flickers is

identified as salient, or increased in salience according to results of the iterative competition process applied to the flicker map.

[0092] A preferred embodiment for a flicker saliency channel hence may include:

- a basic rectified flicker extraction module based on taking the absolute value of the difference between two successive frames.
- its application to several spatial scales and in several image modalities (luminance, chrominance, etc).
- the application of a non-linear spatial competition for salience mechanism within each resulting flicker map.
- the summation of all the maps for a given modality (and several scales) into one summary map for that modality.
- the application on those summary maps of the non-linear spatial competition process
- the summation of all summary maps
- the application on the final result of the non-linear spatial competition process.

[0093] The above system evaluates saliency based on images obtained using a classical red-green-blue representation. This gives rise to two opponency channels (red-green and blue-yellow), an intensity channels, and four orientation channels. These seven channels are processed in separate computational streams. This

can be extended to many more channels. Such multi-spectral or hyper-spectral image sensors may include near and far infra-red cameras, visible light cameras, synthetic aperture radar and so on. With images comprising large numbers of spectral bands, e.g., up to hundreds of channels in some futuristic military scenarios, significant redundancies will exist across different spectral bands. The saliency system can therefore be used to model more sophisticated interactions between spectral channels.

[0094] This may be achieved by implementing connections across channels whereby each feature map at a given scale can receive multiplicative excitatory or inhibitory input from another feature map at the same or different spatial scale. These connections extend the interactive spatial competition for salience already implemented in the saliency model: at each time step, spatial interactions within each map may be iterated, followed by one iteration of interactions across maps.

Supervised training algorithms can be applied to include training of the weights by which the different channels interact. The resulting system may be able to exploit multi-spectral imagery in a much more sophisticated manner than is currently possible.

[0095] The above has described different ways of carrying out detection of the image saliency. The importance is that when exposure to an image or a sequence of images is short, attention

of the (casual) observer is primarily deployed autonomously onto the locations which are most perceptually salient. A close approximation to the perceptual salience at every visual location allows a designer to optimize their work for notice by users. One application of such detection is in the field of advertising. It has been noted by the inventors that users do not perceive all components of a visual environment to be equally interesting. This may be used to evaluate the effectiveness of an advertising model. Hence, the embodiment is described which uses a computer to provide an automatic, objective, and quantitative tool by which the impact of advertising designs can be tested. This may be used on any image, moving or static, including, for example, web pages, billboards, magazine covers, TV commercials, or any medium to which the target audience may be briefly exposed.

[0096] This may be used to calculate saliency/conspicuity of items which are being displayed, for example, in an advertising context. This may include advertisements, visual art and text in print (magazines, news-papers, journals, books); posters, billboards and other outside, environmental displays; advertisements, visual art and text in electronic format on the world-wide-web or on computers; as well as the saliency/conspicuity of dynamic advertisements, visual art and clips in movies, TV film, videos, dynamic display boards or graphical user interfaces. It may also

be used for the saliency/conspicuity of displays of products placed in shop windows, department stores, aisles and shelves, printed ads and so on for product placement. That is, given a particular product (e.g. a soda brand, wine bottle, candy bar), the software evaluates its saliency within the entire display by taking account of the entire view as would be seen by a casual observer or shopper.

[0097] The software can also determine how to change the visual appearance of the product, including its shape and its label, in order to increase its saliency. It can do so by providing specific information to the user on which features, at which spatial scales, are more or less salient than the object or location that the user wishes to draw the attention of the viewer to. For instance, say the user wishes to draw the eye of the viewer to a specific brand of candy bars in an array of candy bars, chocolates and other sweets. By inspecting the conspicuity maps for color, orientation and intensity (see Fig. 1), the user can get a first impression of which objects in the scene are salient because of an intensity difference, because of a color difference or because of their spatial orientation relative to the background. Further information can be provided by having the user inspect the entire pyramid for the different color and orientation maps. Now the user can discover at what particular

TOP SECRET//DO NOT DISTRIBUTE

spatial scale any one object or location in the image is most salient. This can then guide how the user should rearrange the candy display (or the ad) in order to maximize the desired object's saliency.

[0098] The above techniques have taught multiple ways of determining which part of the many maps representing the image has a maximum salience. This can be done from features, feature dimensions, and evaluation of the features at multiple spatial scales. The techniques of increasing the salience effectively uses a search process through parameter space. For example, each parameter may be varied in each direction to determine if that part of the image becomes more salient or less salient. A part of the image, for example, could be made a little redder. Once doing so, an evaluation of whether the saliency increases is made. If the saliency does increase from that change, then the image can be made redder still. This can be continued until the maximum saliency from that parameter is obtained. By carrying out a search process through parameter space, different parts of the image can be made more or less salient. The search process can be carried out through feature channels including any of the feature channels noted above, and through different scales. The parameter is changed systematically throughout each of these

0098-0001

values to determine the effect on saliency, allowing the saliency of different parts of the image to be manipulated.

[0099] An additional aspect learns from the way in which images are made more salient. From this "experience", the system may use a conventional learning system to write rules which say, in a certain kind of the image/background/space, do a certain operation in order to increase the salience of the image portion. This automated system hence provides rules or preferences which can increase the salience.

[00100] Applications within the military, intelligence and security community which require (semi)-automatic evaluation of salient images to detect construction activities, evidence for burials, missile tests, intruders alert or the detection of humans in surveillance applications who behave "conspicuously" are also envisioned. In particular, this application may include multi-spectral images (where not only three color channels but possibly hundreds of different spectral images are combined in a similar manner to yield saliency) as well as moving imagery.

[00101] For applications in the robotic domain, the software can alert (semi)-autonomously moving robotic device of salient locations in the environment that need to be further inspected by a high-resolution sensory system or by a human observer.

[00102] This model may predict where casual observers will place their attention. For example, this could either be done as a service, where ad people send their ad to the service, and the service analyzes it and sends it back with an analysis of its saliency. Another paradigm is a web-based service where people submit images and the software automatically determines the first, second, third etc. most salient locations. The paradigm can also be carried out on a computer such as a PDA with attached camera. The software runs on this hand-held device as a sort of "saliency meter" for determining the saliency of, for example, a product display.

[00103] Other embodiments are within the disclosed invention.